ChinaXiv合作期刊

一种基于微信小程序在RTS2的控制框架扩展[1]

梁　波，田智雁，王　锋，邓　辉，卫守林

（昆明理工大学 云南省计算机技术应用重点实验室，云南 昆明　650500）

摘要：望远镜自主控制是现代天文观测技术的重要组成部分，在当前主流的自主控制系统中，开源的RTS2具备模块化和即插即用的设计理念，且具有快速响应能力和稳定工作的特点，被广泛应用于天文望远镜自主控制系统。由于RTS2基于Linux平台，主要基于CLI进行远程访问控制，所以对观测人员的要求比较高。深入分析RTS2系统，对JSONAPI进行适度改造，以JSON为数据传输格式，以移动终端的微信应用作为载体，跨越不同平台对RTS2天文望远镜控制系统进行数据访问和功能调用。利用微信小程序，将控制系统移植到微信小程序中，使天文技术研究人员能够方便快捷地利用移动终端在微信平台上远程控制天文望远镜和实时监控天文望远镜自主控制系统的状态。采用该模式，可扩展到ASCOM等其他自主控制架构，从而实现一个通用的基于微信小应用的移动终端远程控制。

关键词：RTS2；JSON；微信小程序

中图分类号:TP311.1　　文献标识码:A　　文章编号:1672-7673(2018)

远程望远镜系统第2版[2]（Remote Telescope System 2nd version , RTS2）是一种基于Linux操作系统的远程天文望远镜控制系统，设计目的是实现全自动的控制天文望远镜。根据RTS2开源性特点，天文技术研究人员可以很容易从GitHub获得源代码，并根据GitHub上的方法在Ubuntu操作系统下安装和配置RTS2。

在RTS2中已经拥有了像rts2-mon等CLI工具对系统进行控制，但是这些工具只能在已经安装了这些工具和依赖RTS2软件包的Linux机器上运行，然而，移动终端的广泛使用，科研人员工作和生活半径增大，在需要移动办公和远程监控需求时，RTS2的自备工具极为不便，甚至有些情况鞭长莫及。

对于天文望远镜的观测监控人员来说，能够实时查看运行状态、报警信息和发送控制请求是很有必要的。近年来，国内在天文望远镜自主观测系统的控制扩展上已出现一些研究成果，在RTS2中可以访问RTS2的rts2-xmlrpc服务，通过8889端口号以网络访问的方式对RTS2进行监控和状态查看[1-4]，并且天文观测技术人员通过改进的XMP-RPC接口，使用XML格式数据，实现了浏览器对RTS2系统进行操作，但是XML格式复杂，传输占用带宽比较大，解析XML需要耗费大量的中央处理器资源，不具备轻量级环境中使用；或者继承Rts2 Client Core类实现一个服务，开发浏览器页面通过WebSocket[3-4]对RTS2进行监控和控制。两种实现，都是希望利用浏览器的轻便解决传统CLI环境的局限，在浏览器中，需要记录访问的URL，对于记忆和使用都比较麻烦，不利于望远镜的监控和控制的使用。

随着移动终端的发展，微信使用十分普遍，如能将望远镜的实时监控和控制界面移植

1[1]基金项目：国家自然科学基金（U1631129,U1531132,11403009,61462053,U1231205,11203011）；云南省应用基础研究项目（2017FB001）资助.

收稿日期：2017-08-20；修订日期：2017-10-10.

作者简介：梁　波，男，工程师. 研究方向：计算机应用，天文仪器与方法. Email:lb@cnlab.net.

通信作者：田智雁，男，硕士. 研究方向：计算机应用. Email: tianzhiyan@cnlab.net.

2 http://rts2.org

chinaXiv:201805.00535v1

到微信小程序中，对于观测人员实时控制望远镜系统将更加简便。本文正是基于此背景展开相应的研究工作，在分析其他已有研究成果的基础上，采用 HTTP 协议与微信小程序进行 JSON 数据交互，从而实现 RTS2 自主控制系统进行监控和控制。

1 微信小程序与 RTS2 的数据交互

1.1 微信小程序

微信(WeChat)[3]是腾讯公司 2011 年推出的一个为智能终端提供即时通讯服务的免费应用程序。随着近几年技术的发展，腾讯公司推出了触手可及、用完即走的微信小程序[5]，它是一种不需要下载和安装就可以使用的应用，用户扫一扫或者搜一下即可以打开应用，用完之后退出即可，无需关闭卸载[6]，可在微信内便捷地获取与传播，具有很出色的用户体验；微信支持多种不同操作系统的移动终端，微信小程序开发者可以不用考虑移动终端是 Android 系统还是 IOS 系统，只需要实现开发所需要的功能即可，但是要想使用微信小程序，用户的微信版本必须升级到 6.5.3 及以上版本。

微信小程序的开发与写 HTML 界面很相似，它的主要开发内容也包括界面 index.wxml、index.wxss 和 index.js，并通过微信小程序提供的接口与开发者的服务器进行数据交互。以微信作为载体的微信小程序与 RTS2 进行数据交互的流程如图 1。

图 1 微信小程序访问 RTS2 流程图

Fig 1 The WeChat Micro-program access to RTS2 flow chart

在微信小程序中，每访问 RTS2 服务一次，都需要先访问微信服务器，所以在做服务访问时需要调用微信小程序封装好的访问方法 wx.request(OBJECT)，然后通过微信服务器与 RTS2 服务进行数据交互，最后将 JSON 数据返回给微信小程序。由于微信小程序提供的接口，对访问的 URL 中不能拥有端口号的限制，要想通过微信小程序访问自己的服务器，先得把 observatoryserver 中的端口号做一个地址映射，将端口号隐藏，这样才能很好地通过微信小程序与 RTS2 进行交互。

1.2 数据处理

微信小程序是通过发送 HTTP 请求与 RTS2 服务进行数据交互的，这一点与网络访问 RTS2 服务一致。根据 JSONAPI 提供的各种命令，微信小程序可以通过 HTTP 地址访问 RTS2 服务，RTS2 根据不同的命令，以 JSON 的格式返回对应的数据，为了避免出现中文乱码现象，微信小程序与 RTS2 服务进行数据交互时，都需要设置数据编码格式为 UTF-8。比如，通过地址 http://observatoryserver/apipath/jsoncommands，加上相应的传入参数，就可以获取 RTS2 返回的 JSON 格式的数据 {infotime:1502882474.90689,uptime:1502882474.052562,focuser: “F0”}，在微信小程序中，通过 jquery 很方便地获取该 JSON 格式的数据，通过代码 obj.infotime 可以获取 1502882474.90689 数据，该数据为时间戳，在做数据处理时，需要对该数据进行时间转换，具体方法可以通过以下代码转换：

```
var day = new Date(parseInt(time) * 1000);
var date = "." + time.split(".")[1];
```

3 http://www.chinastor.org/GuoNeiXinWen/8699.html

chinaXiv:201805.00535v1

chinaXiv:201805.00535v1

```
time = day.toLocaleDateString().replace(/\//g, "-") + "T"
+day.toTimeString().substr(0, 8)+ parseFloat(date).toFixed(3);
```

由于不同 JSON 命令，获取的数据不同，在微信小程序里写了一个通用的方法，将 RTS 2 返回的 JSON 格式数据，做了一个基本的转换，将原键值对的数据分开，形成新的 JSON 格式的数据，具体核心代码如下：

```
var json = "{'list':[";
var i = 0;
for(var k in devices){
    if(i>0)
        json +=",";
    json +="{'name':'"+k+"','values':'"+devices[k]+"'}";
    i++;
}
json +="]}";
var obj = eval('('+json+')');
```

前诉 JSON 数据经上述代码转化为{list:[{key: infotime ,value: 1502882474.90689},{ key: uptime,value: 1502882474.052562},{ key: focuser,value: “F0”}]}这样的格式，JSON 格式易于阅读和编写，同时也易于机器解析和生成，并有效地提升网络传输效率。

2 微信小程序设计

2.1 JSON API 命令

根据 GitHub 中最新 RTS2 源码的分析，提供了很多命令可以通过浏览器或移动终端设备获取 RTS2 中的设备信息，并通过一些参数可以控制设备，表 1[4]列出了在 JSON API 中提供的命令。

表 1 JSON API 提供的命令

Table 1 The JSON API provides commands

change_constraints	change_script	cmd	cnst_alt
cnst_alt_v	cnst_time	consts	create_target
dec	devbytype	deviceinfo	devices
executor	expose	get	hasimages
night	obytid	plan	runscript
inc	labellist	labels	lastimage
message	satisfied	script	selval
set	status	sunalt	taltitudes

表 1 提供了很多命令，这些命令可以对天文望远镜自主系统进行远程控制和远程查看设备控制、状态信息，同时还可以获取相应的天文图片。接下来介绍一些主要命令的作用也是本文使用的命令。

（1）控制命令

a、直接控制 cmd

通过使用 cmd 命令，可以根据传入的 device 和 target 两个参数，向 RTS2 中的不同设

4 http://rts2.org/wiki/doku.php?id=code:json

备发送命令控制设备。比如，要观察目标8的下一个目标，通过地址 http://observatoryserver/apipath/cmd?d=EXEC&c=next%208 就可以向设备发送控制命令（在这种情况下，脚本有一个替换为'%20'的空格以符合 http urls 标准）。

b、运行脚本控制 runscript、killscript

微信小程序可以通过 runscript 命令运行脚本、killscript 关闭正在运行的脚本，通过微信小程序向 RTS2 发送控制信息 runscript?d=C0&s=E%201 可以运行已经准备好的脚本（在这种情况下，脚本有一个替换为'%20'的空格以符合 http urls 标准）。

（2）devbytype

通过调用 devbytype 方法，可以通过选择设备类型，查询与之匹配的设备列表，表2[4]列出了可供选择设备类型代码。

表2 设备类型

Table 2 Equipment type

2	MOUNT	8	PLAN	14	AUGERSH	24	INID
3	CCD	9	GRB	15	SENSOR	25	LOGD
4	DOME	10	FOCUS	20	EXECUTOR	26	SCRIPTOR
5	WEATHER	11	MIRROR	21	IMGPROC		
6	ROTATOR	12	CUPOLA	22	SELECTOR		

（3）devices

微信小程序通过使用 devices 命令，可以获取连接到 RTS2 中的所有设备列表，该数据以 json 格式描述。

（4）get

微信小程序通过 get 命令，结合设备名称参数，获取某个连接设备的具体数据，比如设备的状态、位置等详细的数据生成 JSON 格式从 RTS2 中返回到微信小程序里，以便观测人员进行解析显示。

（5）lastimage、currentimage

微信小程序通过 lastimage 命令获取最后采集的图片，通过 currentimage 命令获取当前采集的图片。

2.2 RTS2 的微信小程序终端控制系统实现

根据上述讨论的技术，在移动终端依托微信小程序，利用开源的 RTS2，在微信小程序上实现了一套基于 RTS2 的望远镜系统状态的查看、接 RTS2 上的设备的状态和信息查看和拍摄图片查看的系统原型。系统以部署在 Ubuntu16.04 上的 RTS 2 为服务，使用系统为 Android6.0 版本的手机且安装版本为 6.5.10 的微信，微信小程序与 RTS2 交互的访问流程如图2。

图2 微信小程序功能图

Fig 2 The WeChat Micro-program function diagram

通过微信开发工具将微信小程序发布到微信上即可使用，微信小程序通过使用 devices 命令获取连接设备列表，根据需要查看设备的状态和信息，设备信息、状态和性能数据的显示界面如图3。

chinaXiv:201805.00535v1

chinaXiv:201805.00535v1



图 3 设备信息、状态和性能

Fig 3 Device information, status and performance

根据图 3 的显示信息可以看出通过小程序发送 JSON 封装的命令，获取连接设备的基本信息，微信小程序自带的调试和性能监控功能可以看出 CPU 占用率是 1%，内存消耗只有 151M，运行小程序对手机硬件的要求不高，同时，由于小程序自身拥有即开即用、不需要下载安装等特点，也不会占用系统的存储空间。在现实使用中，观测人员通过微信小程序可以随时随地查看连接设备信息和拍摄的天文图片，同时也可以根据 JSON 命令控制设备。

3 结束语

RTS2 作为一个开源的天文望远镜控制系统，在天文领域许多研究人员对其进行研究和使用，基于浏览器通过 XMLRPC[1]/WebSocket[3-4]协议对其进行实时的控制和状态查询进行扩展，通过微信的广泛使用，以此为载体，提供微信小程序，只需要观测人员或者值守人员拥有微信客户端，使用场景中与手机 OS 无关，可以在移动终端中随时随地查看连接设备状态和获取望远镜拍摄的图片；虽然本文的自主控制系统是 RTS2，同样可以借助 JSON Proxy Server 扩展到 ASCOM 框架中，实现一个基于微信小程序针对自主望远镜控制系统远程的通用框架。

总体来说，本文实现的通用望远镜控制微信程序设计能够满足对连接设备信息查看和控制的观测需要。由于本文基于 HTTP 协议进行，所有获取都是主动模式，与 WebSocket 相比，实时方面有所欠缺，在下一步工作中，可寻找技术手段进行改进。

参考文献：

[1] 冉凡辉,邓辉,梁波,等. 基于 XML-RPC 的 RTS2 自主观测系统远程访问技术[J]. 天文研究与技术—国家天文台台刊,2013,10(4)：372-377.

Ran Fanhui,Deng Hui,Liang Bo, et al. A study of remote access techniques for an RTS2 autonomous observation software system based on the XML-

ChinaXiv合作期刊

RPC[J/OL]. Astronomical Research & Technology—Publications of National Astronomical Observatories of China, 2013,10(4):372-377.

[2] 谌俊毅,陈东,范玉峰,等.基于嵌入式系统的望远镜远程控制实验[J].天文研究与技术—国家天文台台刊,2007,4(1):36-41.

Chen Junyi,Chen Dong,Fan Yufeng,et al. The remote control experiment of the telescope based on the embedded Internet[J]. Astronomical Research & Technology—Publications of National Astronomical Observatories of China , 2007, 4(1):36-41.

[3] 许骏，金振宇.WindowsSockets 实现天文终端的远程控制[J].云南天文台台刊,2001(1):40-46.

Xu Jun,Jin Zhenyu. Remote control of astronomical terminals based on WinSockets[J]. Publications of the Yunnan Observatoty, 2001(1):40-46.

[4] 卫守林,曹子皇,王锋,等.基于 WebSocket 的 RTS2Web 控制研究[J].天文研究与技术—国家天文台台刊,2014: 11（4）: 404-409.

Wei Shoulin,Cao Zihuang,Wang Feng, et al. A study of web control of an RTS2 system based on the WebSocket[J]. Astronomical Research & Technology—Publications of National Astronomical Observatories of China，2014，11（4）：404-409.

[5] 匡文波, 李芮, 任卓如. 微信小程序面面观[J]. 新闻论坛, 2017(2):15-18.

Kuang Wenbo,Li Rui,Ren Zhuoru.Face view micro-program[J]. News Tribune, 2017(2):15-18.

[6] 王天泥.当图书馆遇上微信小程序[J].图书与情报,2016(6):83-86.

Wang Tianni. When library encounters WeChat mini-apps[J]. Library and Information, 2016(6):83-86.

A Control Framework for RTS2 Based on the WeChat Micro-program

Liang Bo, Tian Zhiyan, Wang Feng, Deng Hui, Wei Shoulin

(Key Lab of Applications of Computer Technology of the Yunnan Province, University of Science and Technology of Kunming, Kunming 650500, China, Email:)

Abstract: The independent control of telescope is an important part of modern astronomical observation technology. In the current mainstream autonomous control system, the open source RTS2 has the characteristics of modularity and plug and play, and has the characteristics of fast response capability and stable operation. Widely used in the autonomous control system. As RTS2 is mainly based on the Linux platform, which is mainly based on the CLI for remote access control, the requirements of the observers are limited. In this paper, the RTS2 system is deeply analyzed, and the JSON API is modified to JSON as the data transmission format. The microcomputer application in the

chinaXiv:201805.00535v1

ChinaXiv合作期刊

mobile terminal is used as the carrier, and the data access and function call of the RTS2 telescope control system are carried out across different platforms. The use of WeChat Micro-program, the control system transplanted to WeChat Micro-program, so that astronomical researchers can quickly and easily use the mobile terminal in the WeChat platform remote control telescope and real-time monitoring telescope autonomous control system status. With this model, it can be extended to other autonomous control architectures such as ASCOM, so as to realize a remote control of mobile terminal based on micro-application.

Key words: RTS2; JSON; WeChat Micro-program

chinaXiv:201805.00535v1